



## Referencia objetos

### Transcripción

[00:00] Ya vimos en la clase anterior cómo agregar comportamiento a nuestros objetos ya creados y ahora vamos a entrar un poco en definir y explicar cómo es que se relacionan los objetos entre sí. Vamos a recordar un poco la clase anterior. Aquí tenemos nuestra entidad cuenta. Nuestra entidad cuenta tiene tres métodos que son depositar, retirar y transferir y habíamos ya acordado que para nuestro banco actualmente ese iba a ser digamos el objeto cuenta, con los campos y sus métodos.

[00:36] Ahora, por reglas del negocio, quizás nosotros necesitemos adicionar otros valores a la cuenta. Vamos al código. Por ejemplo, si sabemos que tiene un titular, en este caso de repente la cuenta también necesita el número de documento del titular, también necesita el número de teléfono.

[01:04] Cuando vemos que ya comenzamos a poner algunos otros campos en cuenta, no sé, de repente algún otro campo, sexo o alguna cosa así, vamos viendo que comienza a quedar ya un poco grande por así decirlo, y aquí tenemos ya que comenzar a pensar más en el dominio de la cuenta. ¿A qué me refiero con esto?

[01:26] Sabemos que la cuenta tiene saldo, la cuenta tiene agencia, la cuenta tiene un número y en efecto tiene un titular, pero será que la cuenta tiene un número de documento, la cuenta debería tener un teléfono, la cuenta debería tener sexo, entonces aquí es donde ya tenemos que comenzar ya a separar nuestros dominios, separar nuestro concepto y este es un claro síntoma de que nosotros ya necesitamos aislar estos campos en otro objeto.

[02:00] Volviendo a la presentación, tenemos aquí los campos que ya hemos adicionado. Y dado que lo que necesitamos es desacoplarlos de cuenta, vamos a crear un objeto cliente. En este caso el objeto cliente va a tener número de documento, va a tener teléfono y va a tener nombre. Perfecto.

[02:24] Yendo al código, ya poniendo en práctica esto, vamos ahora mismo a crear nuestro objeto cliente. Damos clic derecho, new class y le llamamos cliente. Listo. En cliente, vamos a copiar todo esto con excepción de este último parámetro que fue solo para fines de ejemplo, y vamos a pegarlos aquí. En efecto ahora ya vemos pues que el cliente tiene nombre, no tiene título, tiene un nombre.

[03:08] El cliente tiene un número de documento. Bueno, es un número pero lo estamos definiendo como string. Esa ya es una discusión para otro momento. Y también tiene un número de teléfono. Estos campos, dado que son informaciones del cliente, podemos llenarlo como un objeto independiente. Vamos a crear una clase de prueba aquí, clase, y le ponemos TestReferencia.

[03:37] Ah, creo que ya tengo un TestReferencia aquí. TestReferencia 2. Listo. Se dan cuenta, Eclipse ya nos dice ya si tenemos una clase con el nombre repetido, nos va a dar el error. Esa es una gran ayuda para nosotros. Sin el tiempo de compilación nos daría un error tremendo. Le damos finish y comenzamos con nuestro método main. Listo.

[04:05] Entonces sabemos que cliente es un objeto nuevo, por lo tanto, ¿qué podemos hacer? Podemos instanciarlo. Entonces vamos a llamarlo cliente y lo vamos a llamar Diego. Y va a ser igual a new cliente. "Ctrl + espacio" para autocompletar, enter, y tenemos ahí a cliente.

[04:24] Y a Diego yo puedo asignarle valores, por ejemplo Diego.nombre. Va a ser su nombre obviamente. Aquí puede ser documento, y aquí le asignamos el número de documento. Punto y coma al final. Y aquí puede ser teléfono, habíamos quedado. Y le asignamos un número de teléfono aquí. Deja ver aquí.

[05:10] Ahora que tengo mi objeto Diego ya creado y ya con los datos, en teoría yo podría crear ya una cuenta, digamos que vamos a llamar ahora una cuenta. Cuenta igual new Cuenta. Y esta cuenta la vamos a llamar cuenta de Diego. Entonces, volvemos aquí y a la cuenta de igual forma podemos asignarle diferentes valores. Por ejemplo cuentaDeDiego.agencia, vamos a ponerle un 1 y podemos seguir así.

[05:59] ¿Cómo es que yo creo la relación entre cuenta y Diego, en este caso entre la cuenta de Diego y Diego? ¿Yo cómo es que voy a unir a ellos dos? ¿Cómo le digo "oye, cuenta de Diego, por si acaso, tu titular es Diego"? Volviendo a la presentación, lo que yo necesito aquí es crear una referencia a este objeto de aquí. Tenemos que recordar siempre esa palabra mágica: referencia. En Java recuerden que todo es en base a referencias.

[06:31] ¿Cómo creo yo una referencia de este objeto, hacia este objeto? Fácil. Vamos aquí a cuenta y recuerden que aquí teníamos string titular. Pues adivinen qué. Esto ya no va a seguir siendo un string. Al igual que en cualquier tipo de variable que declaramos en Java., a la izquierda tenemos siempre el tipo de dato que va a ser esa variable. Adivinen qué tipo de dato va a ser mi titular.

[07:05] Exacto, va a ser un tipo de dato cliente. Entonces, la cuenta está referenciando aquí a un cliente que va a ser su titular, por lo tanto si yo aquí asigno el valor de agencia, accediendo al campo declarado en el objeto cuenta de esta forma, yo podría hacer algo así como cuentaDeDiego.titular. Y titular recibe un objeto ¿de qué tipo? Tipo cuenta.

[07:45] Y casualmente tengo un objeto tipo cliente aquí, perdón, no tipo cuenta, tipo cliente. Casualmente yo tengo aquí un objeto del tipo cliente, entonces referenciarlo sería algo tan fácil como, a ver aquí, tan fácil como pegar aquí, punto y coma. Vemos que el código compila, y yo podría hacer ahora algo así como cuentaDeDiego.titular. Y a través de la referencia al titular yo tengo acceso a todos los atributos del objeto cliente. ¿Se dan cuenta aquí?

[08:34] Yo entré a través del objeto cuenta, ahora estoy en el objeto cliente y ya tengo acceso a todos sus atributos. En este caso voy a imprimir el documento por ejemplo. Guardamos todo, ejecutamos, a ver. Ejecutamos el método. Run. Abrimos nuestra consola y en efecto vemos que él ha impreso pues el número de documento que yo grabé aquí.

[09:09] Ahora, para ir un poco más a profundidad con este concepto, si yo hiciera por ejemplo, `system.out.println`, a ver. Algo así como `cuentaDeDiego.titular`. Si yo imprimiera solamente el titular, yo estaría accediendo al objeto titular. Y si ustedes recuerdan, dado que titular referencia a un nuevo objeto cliente que está aquí, ya sabemos que todo nuevo objeto crea una ruta en la memoria de Java.

[09:54] Si yo imprimo este valor de aquí, entonces él me va a dar aquel código extraño, aquella referencia extraña, esa es la dirección de memoria en la cual está este titular. Y adivinen qué. Si yo imprimo directamente, vamos a usar el `sout`, directamente al objeto Diego, guardamos, imprimimos, es el mismo objeto al cual Java está haciendo referencia.

[10:36] Por lo tanto, repasando los conceptos que ya aprendimos anteriormente, como es una referencia, cualquier cambio que yo haga aquí, va a reflejarse directamente en ese objeto que está dentro de titular y que a la vez está dentro del objeto cuenta.